



# Hybrid pyramid-DWT-SVD dual data hiding technique for videos ownership protection

Farhan Alenizi<sup>1</sup> · Fadi Kurdahi<sup>2</sup> · Ahmed M. Eltawil<sup>2</sup> · Awad Kh. Al-Asmari<sup>3</sup>

Received: 18 October 2017 / Revised: 14 September 2018 / Accepted: 24 September 2018 /  
Published online: 9 November 2018  
© Springer Science+Business Media, LLC, part of Springer Nature 2018

## Abstract

This paper proposes a hybrid pyramid Discrete-Wavelet-Transform (DWT) Singular-Value-Decomposition (SVD) data hiding scheme for video authentication and ownership protection. The data being hidden will be in the shape of a main color logo image watermark and another secondary Black and White (B&W) logo image. The color watermark will be decomposed to Bit-Slices. A pyramid transform is performed on the Y-frames of a video stream resulting in error images; then, a Discrete Wavelet Transform (DWT) process is implemented using orthonormal filter banks on these error images, and the Bit-Slices watermarks are inserted in one or more of the resulting subbands in a way that is fully controlled by the owner; then, the watermarked video is reconstructed. SVD will be performed on the color watermark Bit-Slices. A secondary B&W watermark will be inserted in the main color watermark using another SVD process. An enhanced detection technique is developed to estimate the Singular Values to reconstruct the original color watermark image. The overall robustness of this scheme is measured when common attacks are applied to the test videos. A main contribution in this research is that the original host video is not required for the extraction process, the good robustness against different attacks specifically compression, transcoding, temporal and geometrical attacks and the duality in the hiding process. The simulation results show that the proposed algorithm achieves well under both the visual and the metric tests. Furthermore, it performed well against intentional and unintentional attacks. The reconstruction was perfect without attacks, while the average Bit-Error-Rates (BER's) achieved under attacks are in the limits of 2% for the color watermark and 5% for the secondary watermark; meanwhile, the mean Peak Signal-to-Noise Ratio (PSNR) is 57 dB.

**Keywords** Watermarking · Filter banks · Data hiding · Pyramid transform · Video coding · Singular value decomposition

---

✉ Farhan Alenizi  
farhan414@gmail.com

# 1 Introduction

The delivery and distribution of various types of multimedia became easier as the new advances in digital communications and information networks are reaching new limits and capabilities every year. These forms of digital information can be easily stolen and exploited when the appropriate precautions are not put in place. These concerns motivated significant research in image and video watermarking fields [23]. Watermarking is a type of data hiding and it is used primarily for authentication and ownership protection. New innovations in video processing techniques, such as compression and transcoding techniques, has brought new challenges to watermarking. For instance, High efficiency video coding (HEVC) or H.265 standard was introduced officially in 2013, it needs on average only half the bit rate of its predecessor, ITU-T H.264 — MPEG-4 Part 10 'Advanced Video Coding' (AVC), which was considered the most deployed video compression standard worldwide [26]. The new standard is expected to be phased in gradually as the new display technologies and networks capabilities outgrow their current limits [25].

Various watermarking schemes that use different techniques have been proposed over the years [5, 10, 11, 13, 17, 19, 22, 29]. To be effective, a watermark must be imperceptible within its host, extracted with ease by the owner, and robust in the face of both intentional and unintentional distortions [6, 11, 14, 18]. Adaptive embedding which depends on image contents and pixel correlations while embedding can be used in image steganography. A compact steganographic embedding function is proposed to ensure the correctness and efficiency, and a pixel correlation function is utilized to discriminate the image smoothness in [15]. The dependencies inside the images are utilized frequently; another medical JPEG image steganographic scheme based on the dependencies of inter-block coefficients is proposed in [16].

Multi-resolution decompositions of images are very popular in the areas of images and videos codings, specifically compression and data hiding [3]. The pyramid scheme, for instance, which was introduced by Burt and Adelson [8] is a form of multi-resolution analysis and proved to be very useful in images compression, it can use linear and nonlinear interpolation and decimation operators. On the other hand, Discrete Wavelet Transform (DWT) is another multi-resolution process; it has wide applications in the different areas of image and video processing such as compression, noise reduction and watermarking [1]; this is attributed to its characteristics in : space-frequency localization, multi-resolution representation and superior Human Visual System (HVS) modeling [19]. Furthermore, the Singular Value Decomposition (SVD) is a powerful technique in many matrices computations. It has the advantage of being more robust to numerical errors [5]; this property of SVD analysis besides others made it useful in image and video watermarking [24, 30]. To have a robust watermarking technique, both the hiding and the extraction processes should be optimized. This in turn necessitates the need for enhanced detection process with no significant extra cost in terms of visual quality or computational complexity.

In this research, our target is to develop a dual watermarking technique using mainly a hybrid pyramid-DWT-SVD analysis. The overall technique will be used for data hiding in encoded videos to meet the requirements of imperceptibility, robustness, security, and computational complexity. The hiding process will be composed of two stages for security reasons; furthermore, a new algorithm will be derived to enhance the extraction and detection processes. A great deal of randomness will be used in many aspects including but not limited to: the filter banks generation and the hiding process to ensure high level of security. The overall performance of the proposed technique will be measured when common aggressive attacks are applied to the test videos. Moreover, special robustness tests against H.264

and H.265 compressions and transcoding are performed to illustrate the performance of our system against these attacks.

## 2 Proposed watermarking technique

### 2.1 Embedding method

In this subsection, we introduce our digital video watermarking technique for the purpose of authentication and ownership protection. The proposed technique is aimed at achieving reasonable degrees of robustness, imperceptibility and security. The embedding technique consists of two stages: the first stage is the decomposition process and the second stage is the hiding process. The hiding process is a dual one. The main watermark that will be used in the videos is a color RGB image; moreover, another smaller B&W watermark will be hidden in this color watermark. The reason for this duality is to establish a high degree of security; furthermore, this would help in generating a built-in random spread spectrum sequence that will be used in the main hiding process. A general illustration of this dual hiding process is shown in Fig. 1.

The watermark can be a logo color image of size  $N * M * 3$  pixels. The encoded videos are primarily in the YUV color space; this space is more efficient in representing the images than the traditional RGB space. The watermarking process can take place in any of the three components Y, U or V. Our proposed algorithm will use the luminance Y frames as host images for the multi-resolution watermarking process; that is, the watermark will be inserted or distributed in one or more of the subbands that result from the hybrid pyramid-wavelet decomposition process. Choosing the analysis and synthesis filters is an important aspect in the efficiency of the reconstruction process. For the wavelet decomposition, special type of filters known as the orthonormal filter banks will be used [27]. These filter banks can be generated randomly depending on the generating polynomials; hence, by generating random numbers for the polynomial coefficients, it's possible to build multiple filter banks that are used for the different stages of our decomposition processes. The filters for the pyramid transform are unconstrained, typically, zero-phase FIR filters are used [27].

The proposed technique starts first by performing a pyramid decomposition on the host Y-frame; further levels can be performed as well. If  $x_0(n_1, n_2)$  is the original image of size  $L1 * L2$  pixels, where  $n_1$  represent the rows and  $n_2$  represent the columns; then the pyramid structure can be done as shown in Fig. 2 [2].

For decimation by a factor of 2, the image will be filtered using analysis lowpass filter  $H$ , and then it will be decimated by a factor of two. This results in an image  $x_1(n_1, n_2)$  which is 1/4 of the size of  $x_0(n_1, n_2)$  and it is called the first-level image of the pyramid. The

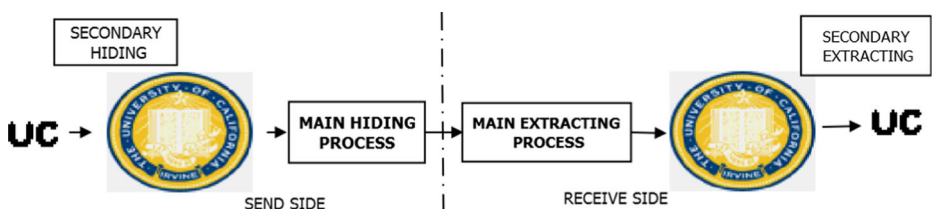


Fig. 1 A general illustration of our dual hiding process

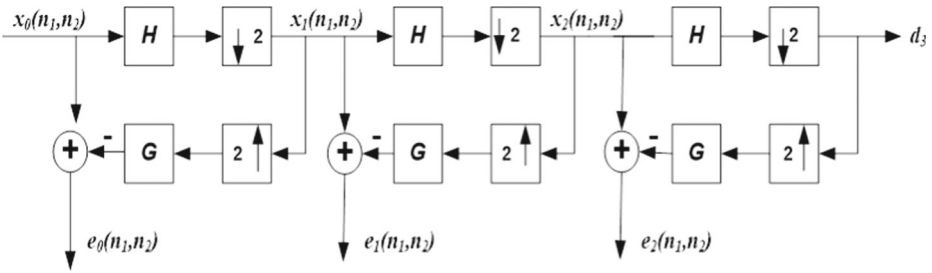


Fig. 2 Three-level pyramid decomposition of an image  $x_0(n_1, n_2)$

second level image  $x_2(n_1, n_2)$  can be obtained from  $x_1(n_1, n_2)$  by the same process, and this process is repeated for the higher levels. The image  $x_1(n_1, n_2)$  can be interpolated by a factor of 2 and then filtered using synthesis filter  $G$ . The resulting image will be  $I[x_1(n_1, n_2)]$ . Where  $I[.]$  is the spatial interpolation and filtering operation. The synthesis filter  $G$  is a time reversal version of the analysis filter  $H$ . The difference (error image)  $e_0(n_1, n_2)$  is given by:

$$e_0(n_1, n_2) = x_0(n_1, n_2) - I[x_1(n_1, n_2)] \tag{1}$$

This process can be done for the higher levels and this will result in the error images  $e_1(n_1, n_2), e_2(n_1, n_2) \dots$  etc. The optimizing of the analysis and synthesis filters plays the major role in the perfect reconstruction of the images. For watermarking purposes, random filters will be used. Quadrature Mirror Filter (QMF9) is an example of optimum filters; it satisfies symmetry, normalization, unimodality and equal contribution. The frequency response of this filter is shown in Fig. 3.

The error image that results from the pyramid decomposition process will be used for our DWT hiding process. The reason for this pre-DWT process is that the resulting error image has broader frequency spectrum than the original image, and hence this would give more space for hiding our watermark. This, in turn, will contribute less visual artifacts to the host

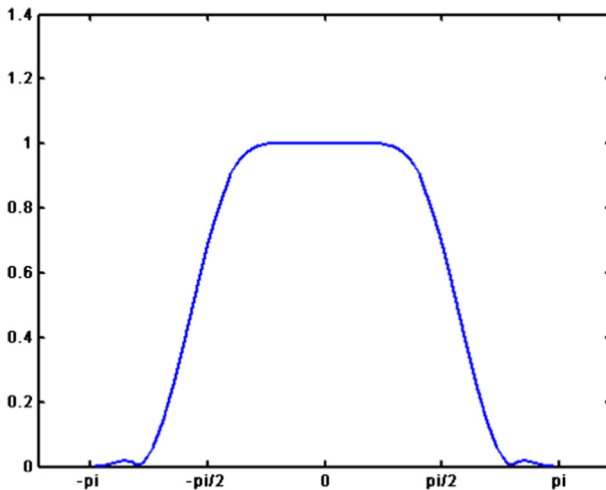


Fig. 3 Frequency response of QMF9 filter

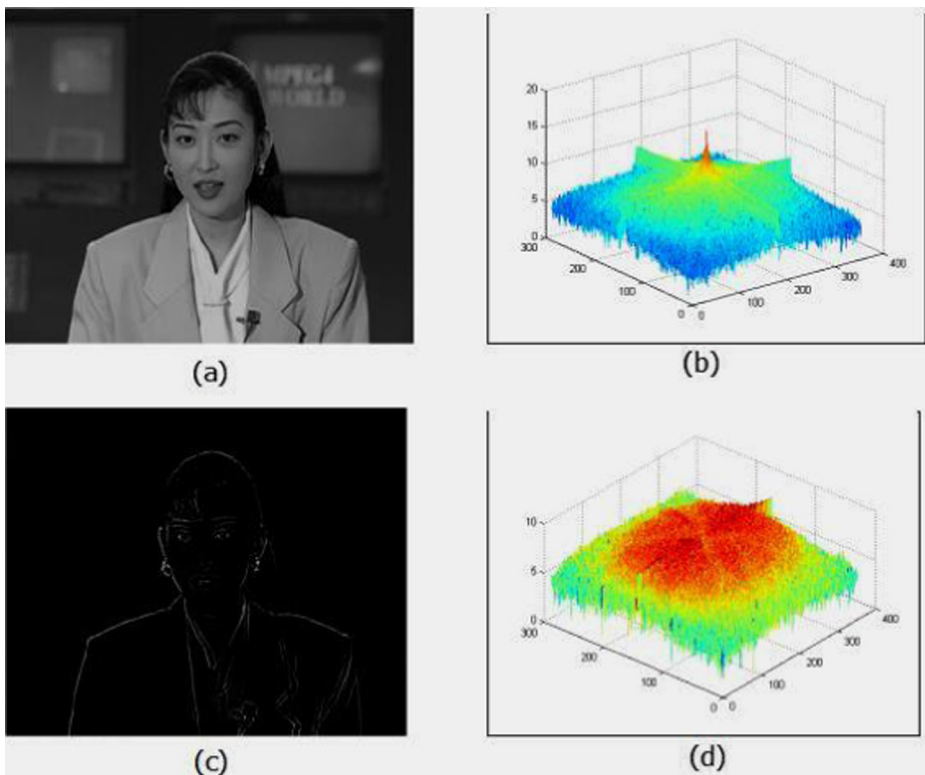
image which is important to achieve the perceptual imperceptibility; this can be shown in Fig. 4.

The DWT orthonormal analysis and synthesis filters can be constructed in such a way that they have large sidelobes. This allows higher energy in the medium frequency bands of the spectrum of the images, to avoid as much as possible the effects of different images' processing techniques that are applied at one stage or another. Depending on the number of the decomposition levels, each filter bank can be used for one level of the DWT decomposition and reconstruction; furthermore, full control on both the structure and the number of levels of the decomposition process can be established to address the security concerns. The bands in the middle frequencies will be used for hiding in general, this in turn would avoid the use of the lower frequency bands, where most of the energy is in, and the higher frequency bands which are susceptible to compression and other image processing attacks such as low-pass filtering. The watermark might be distributed in many subbands; this scenario is helpful in counteracting the Non-linear Collusion Attack. However, to find the best band to hide in, the *directive contrast* will be used as will be shown later in this research.

Singular Value Decomposition (SVD) was used extensively in images and video watermarking; the image  $A$  can be decomposed according to this relation:

$[U S V]=SVD(A)$  where:

$$A = U * S * V^T \quad (2)$$



**Fig. 4** **a** the original image **b** the frequency spectrum of the original image **c** the error image  $e_0$  **d** the frequency spectrum of the error image  $e_0$

$U$  and  $V$  are orthogonal matrices, while  $S$  is a diagonal matrix contains the singular values; these singular values are distributed in a descending order. Like the eigen values analysis, the greatest values of the singular values comprise the greatest amount of information in the decomposed matrix; this is the reason that SVD can be used in images' compression processes. Due to numerous image processing attacks, especially aggressive compression, these singular values could change dramatically, which would affect the reconstructed matrix. Most of the watermarking techniques that use the singular values decomposition depend on hiding the singular values of the watermark in the host image; these methods, however, usually require the original image during the extraction process, hence these methods are semi-blind methods [4, 5, 7, 24]. Our proposed method does not require the original images or video frames; the watermark will be hidden in the host image using the pyramid-wavelet hiding process and then in the extraction process, an approximated watermark will be estimated, and by performing other processing and doing singular value decomposition, optimal singular values can be established that have the minimum root-mean-square error to the original singular values. These established singular values can be used in establishing the reconstructed watermark.

The watermark used in our technique is an RGB color image; the three components: the Red, the Green and the Blue will be extracted, then Bit-Slicing process will be performed on each component, i.e, the slices corresponding to the bits from the least significant to the most significant will be established. Since the most significant bits (from 5 to 8) contain most of the information, these bit-slices will be used only [5]. The information contained in this prospect are the Luminance and Color components of the watermark. This is illustrated in Fig. 5; these individual binary slices will be used as separate watermarks in the hiding process, and then, they will be reconstructed afterwards.

A method to embed the binary watermark using pseudo-random sequence is proposed in [21]. This method establishes the watermarking embedding process by converting the original watermark image  $Q$  to a binary sequence  $S$  of length  $M$  where the data pixels are valued as +1 and the background pixels are valued as -1. Moreover, a pseudo-random sequence  $P$  that has the same length  $M$  as the watermark sequence is generated using a secret key and, similarly, is represented as binary bits that are valued either +1 or -1. The DWT coefficients of the subbands which will be used for the embedding process can be represented as a matrix  $Q_1$  of the same size as the watermark, and it can be converted to a vector  $T$  of length  $M$ . The watermark is embedded into the vector  $T$  to obtain a new vector  $T'$  according to the following rule:

$$t'_i = t_i + \alpha * p_i * s_i, \text{ for } i = 1, 2, \dots, M \quad (3)$$

where  $\alpha$  is a magnitude factor which is a weighting constant that controls the strength of the processed watermark. The value is selected to offer a trade-off between robustness and visual quality. Furthermore, choosing the weighting factor should take into account many issues such as the compression ratio, the smoothness of the image, and the detection process. Moreover, the energy content in the wavelet subbands should be taken into account; one way to get the magnitude factor is to compare the original coefficients of the host DWT subband  $Q_1$  and that of the original watermark image  $Q$  according to this empirical formula:

$$\alpha = 2 * \sqrt{\frac{E(Q_1)}{E(Q)}} \quad (4)$$

where  $E(Q_1)$  denotes the energy of the original wavelet coefficients, while  $E(Q)$  denotes the energy of the watermark matrix  $Q$  which are the sum of the square elements. The enhanced

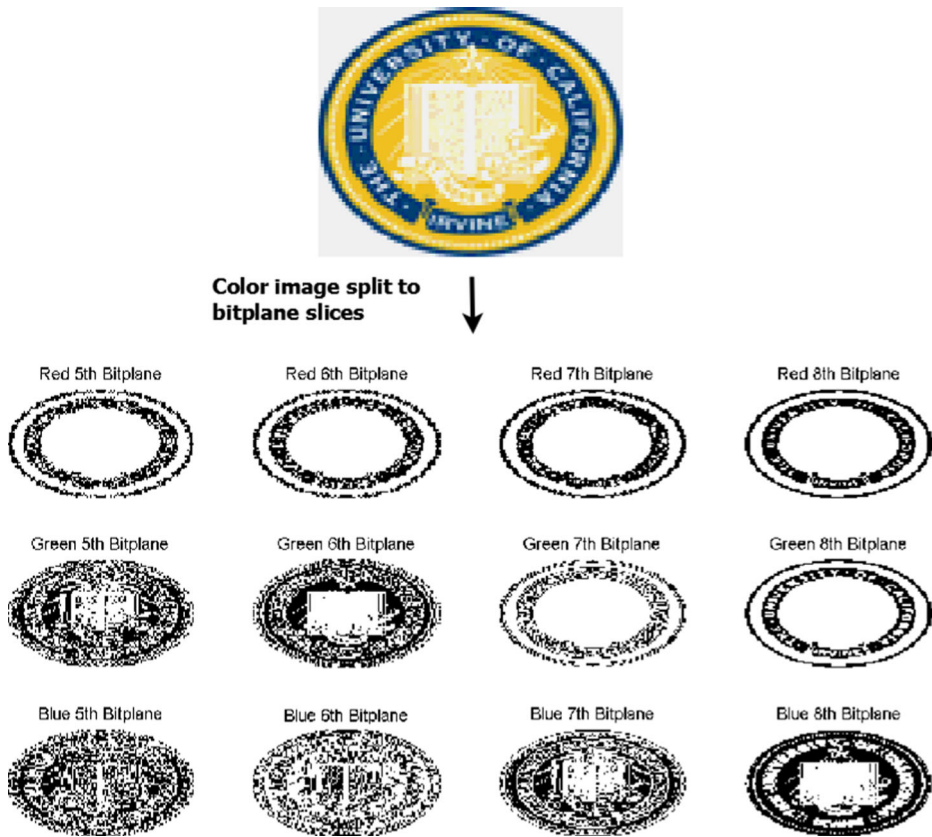


Fig. 5 Color image and its most significant bit-slices

wavelet coefficients are used then in their respective places to reconstruct the watermarked frame. The overall hiding process with color watermarks is shown in Fig. 6. It can be shown, and this depends on the decomposition structure, that the Low-Low frequency version (LL) of the decomposed image is avoided since this sub-band contains most of the information of the original image; the other images represent the Low-High (LH), High-Low (HL) and High-High (HH) bands, and they can be used for hiding.

### 2.2 The directive contrast

To choose the best band for hiding, *directive contrast* can be used [7]. The various directive contrasts for any DWT decomposition level  $i$  are defined as:

- Horizontal Contrast:  $C_i^H = LH_i/LL_i$
- Vertical Contrast:  $C_i^V = HL_i/LL_i$
- Diagonal Contrast:  $C_i^D = HH_i/LL_i$

Directive contrast depicts the high frequency information of an image and the relative intensity of high frequency to the background. To choose the best band to hide in, the band with the highest directive contrast can provide the highest capacity. One way to compare is to use

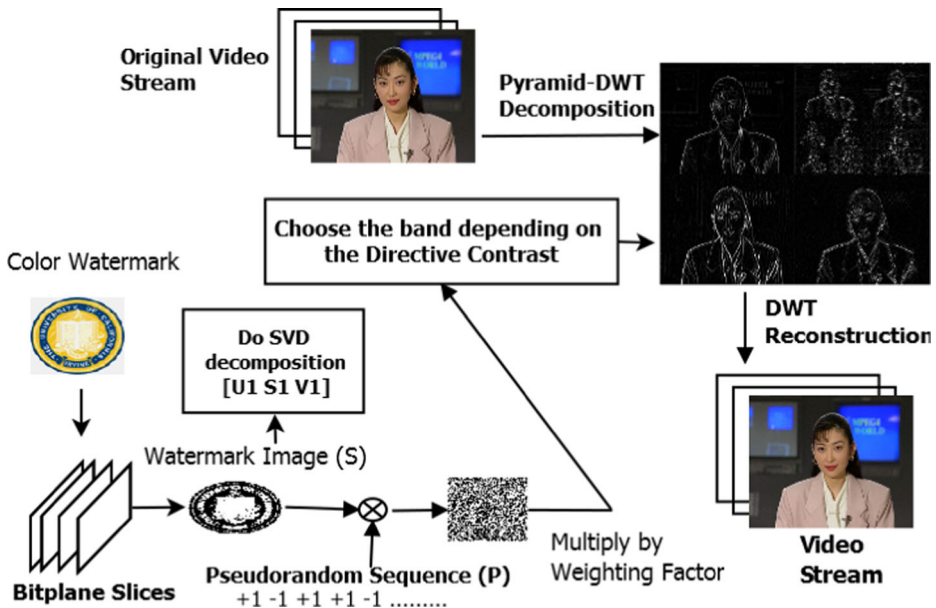


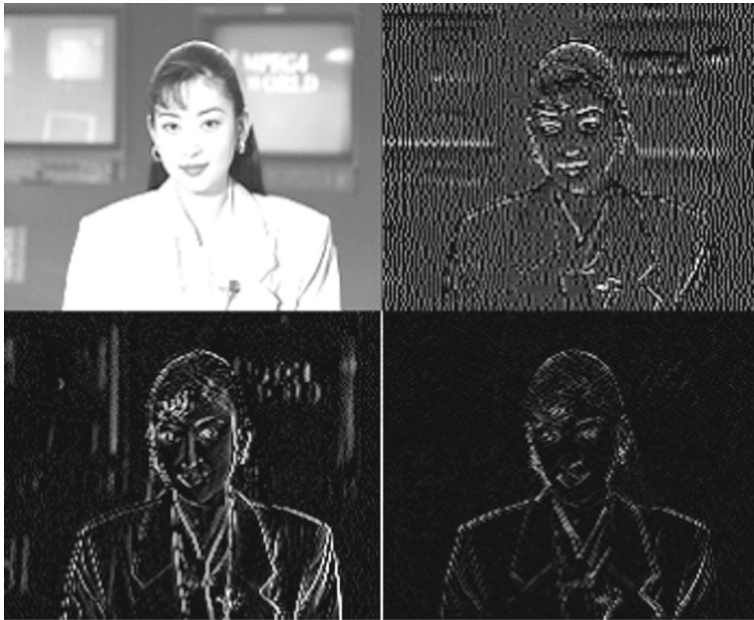
Fig. 6 The main color watermark hiding process

the norm of matrices; this can be shown in Fig. 7 where one level of DWT was performed on the Akiyo video; It can be shown that *HL* band (the left bottom one) is the band with the highest contrast, and so it is the best band for hiding; experimental results are accordant with these remarks.

### 2.3 1D discrete fourier transform

Security issues always arise when designing a robust and reliable hiding system. Applying a fixed watermark to each frame in the video leads to the problem of maintaining statistical invisibility [31]. Moreover, applying independent watermarks to each frame also presents a problem if these frames have few or no motion regions; these motionless regions in successive video frames may be statistically compared or averaged to remove independent watermarks. Attacks of such natures and scopes are normally referred to as collusion attacks. The Inter-frame collusion attacks, for instance, exploit the inherent redundancy in the video frames or in the watermark to produce an unwatermarked copy of the video; these attacks can be divided to Watermark Estimation Remodulation (WER) attack, and Frame Temporal Filtering (FTF) attack [28]. Classifying the video frames into motion and motionless frames is useful in this regard. The motion issue, in fact, is a relative one; since most of the time there is some sort of motion in the videos, but what interest us here are: the amount of the motion, how fast is the motion, and the distribution of this motion all over the space domain of the frames. Most of the video compression techniques use Inter-frame motion estimations to encode the frames; however, a useful and simpler method other than these to detect static and dynamic scenes in videos can be developed using the *1D Discrete Fourier Transform* (DFT). The 1D DFT in temporal direction transforms a group of pictures (GOP) into a temporal frequency domain; in this domain, the spatial information and temporal frequency information exist in the same frame. Higher frequencies correspond to the fast motion from



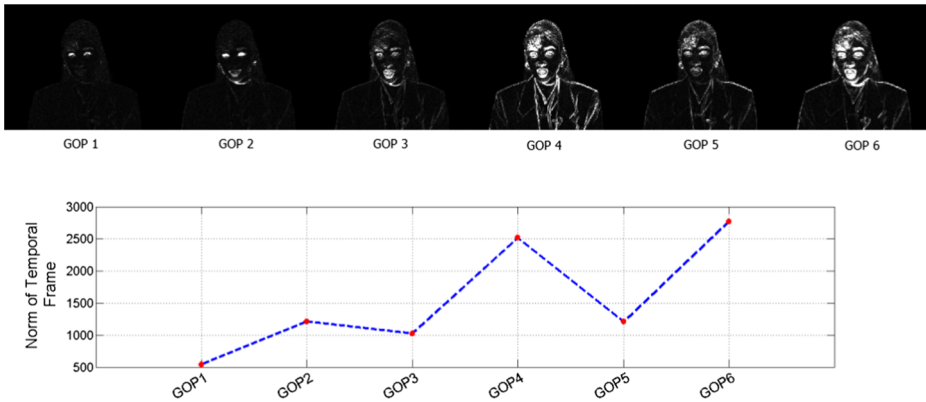


**Fig. 7** 1-level DWT of Akiyo Y-frame showing the directive contrasts of the bands

one frame to other frames [17]. The 1D DFT of a video  $f(x,y,t)$  of size  $M \times N \times T$ , in which,  $M \times N$  is the size of each frame and  $T$  is the total number of frames in the Group of Pictures (GOP), is given by:

$$F(u, v, \tau) = \sum_{t=0}^{T-1} f(x, y, t) e^{-j2\pi(\tau t/T)} \quad (5)$$

where  $u$  and  $v$  represent the spatial domain while  $\tau$  represent the temporal domain. Taking the GOP as 5, a series of spatio-temporal frames can be established for the Akiyo video. 30 frames of Akiyo video were transformed using the 1D DFT, and since the DFT is a symmetric process in one GOP, so it is sufficient to show the first spatio-temporal frame of each group of pictures. Figure 8 shows the 6 temporal frames of Akiyo video that correspond to the original 30 frames, and their norms. The edges shown in these frames represent high frequencies which correspond to motion in temporal domain, and the distribution of this motion in each frame, and hence the value of the norms represent the amount and speed of motion in each group of pictures; for instance, the intensity of the edges shows the blinking of the eyes and the movement of the head or the whole body; it can be seen that the background is motionless which is what we expect. Setting a threshold that classify video frames into dynamic and static frames can be done, and this would help us in establishing a hiding process that is more secure and reliable and can counteract the aforementioned collusion attacks. Depending on this analysis, different watermarks will be hidden in motion frames and the same watermark will be hidden in motionless frames. In fact, our color watermarks were split into bit-slices and this solved the problem in motion frames partially; furthermore and since our watermarking process is a dual one, and the hiding process takes place in transform domain rather than the space domain, and the bands being used are not confined to the high frequency ones, the effect of averaging and collusion attacks is reduced as well. This will be shown in the experimental results later on. Using 1D-DFT to establish motion



**Fig. 8** The 1D DFT of 6 GOP's of 30 frames of Akiyo video and their corresponding norms

information is not the only way that can be used. 3D DWT, for instance, can be used to establish Spatio-temporal components of videos. Choosing the best way to establish motion in frames depends mainly on the application and other factors such as computational complexity. Since we are concerned only with estimating motion but not in precise way, using 1D-DFT is sufficient at this stage.

The pseudo code of the hiding process is illustrated down.

### Hiding stage

1. Convert the color watermark to 12 bit-plane slices  $W_j, j=1,2 \dots 12$
2. Perform SVD on the bit-slices,  $[U_1 \ S_1 \ V_1] = SVD(W_j)$ .
3. Multiply the bit-plane slice by a pseudorandom sequence ( $P_j$ ):  $W'_j = W_j * P_j$
4. Multiply the scrambled bit-slice by a weighting factor  $k$ ,  $W''_j = k * W'_j$
5. Read input video frames  $F_i, i=1,2 \dots n$ 
  - For  $i=1:n$  {
  - Get the YUV components.  $F_i \Rightarrow Y_i, U_i, V_i$
  - Perform 1D Discrete Fourier Transform (1D-DFT) on  $Y_i$  frames.
  - Perform Pyramid decomposition on the  $Y_i$  frame.
  - Perform one or two levels of DWT on the error image  $e_0$  based on the video resolution.
  - Use 1D-DFT values to choose groups of pictures (GOP) for hiding.
  - Use Directive Contrast (DC) to find the best DWT-subband for hiding, where: {Choose the band with Max(DC)}.
  - Hide the scrambled watermark  $W''_j$  in the band found in previous step.
  - Perform Pyramid and DWT reconstruction of the modified  $Y'_i$  frame.
  - Reconstruct the  $Y'_i, U_i$ , and  $V_i$  components to get the watermarked video stream  $F'_i$
6. Store or Transmit.

### 3 Secondary hiding process

As mentioned beforehand, this watermarking process is a dual one, so smaller binary watermarks can be hidden in the Red, Green and Blue components of the color watermark. A

method getting benefits of some properties of the Singular Value Decomposition in Grey scale images was proposed in [24]. The method which is shown in Fig. 9 depends on dividing the image into 4x4 smaller matrices, then SDV process is performed on each matrix; this would result in the orthogonal and singular values matrices U, S and V. By taking the U matrices, it was found that there is big correlation between the elements of the first columns of these matrices, so two other matrices named  $M_1$  and  $M_2$  can be established using the 3rd and the 4th elements of the first columns of the  $4 \times 4$  U matrices. Experimental tests showed that these two elements provided the greatest correlations, but this does not exclude the possibility that other elements could be used. A binary watermark  $W$  of the same size of  $M_1$  can be hidden using this relation:

$$\text{if } W = 1, \begin{cases} u'_{31} = \text{sign}(u_{31}) * (u_{avg} + T/2) \\ u'_{41} = \text{sign}(u_{41}) * (u_{avg} - T/2) \end{cases} \quad (6)$$

$$\text{if } W = 0, \begin{cases} u'_{31} = \text{sign}(u_{31}) * (u_{avg} - T/2) \\ u'_{41} = \text{sign}(u_{41}) * (u_{avg} + T/2) \end{cases} \quad (7)$$

where  $u_{avg}$  is the average of  $u_{31}$  and  $u_{41}$ , and  $T$  is a weighting factor. The whole secondary hiding process for RGB color watermarks is shown in Fig. 9. The resulting color watermark can be used for the main hiding process shown in Fig. 6. Chaos mapping in the shape of *Arnold Transform* was used to scramble the watermark before hiding for security reason. The resulting chaos mapped matrix can be used as a pseudo-random sequence for the

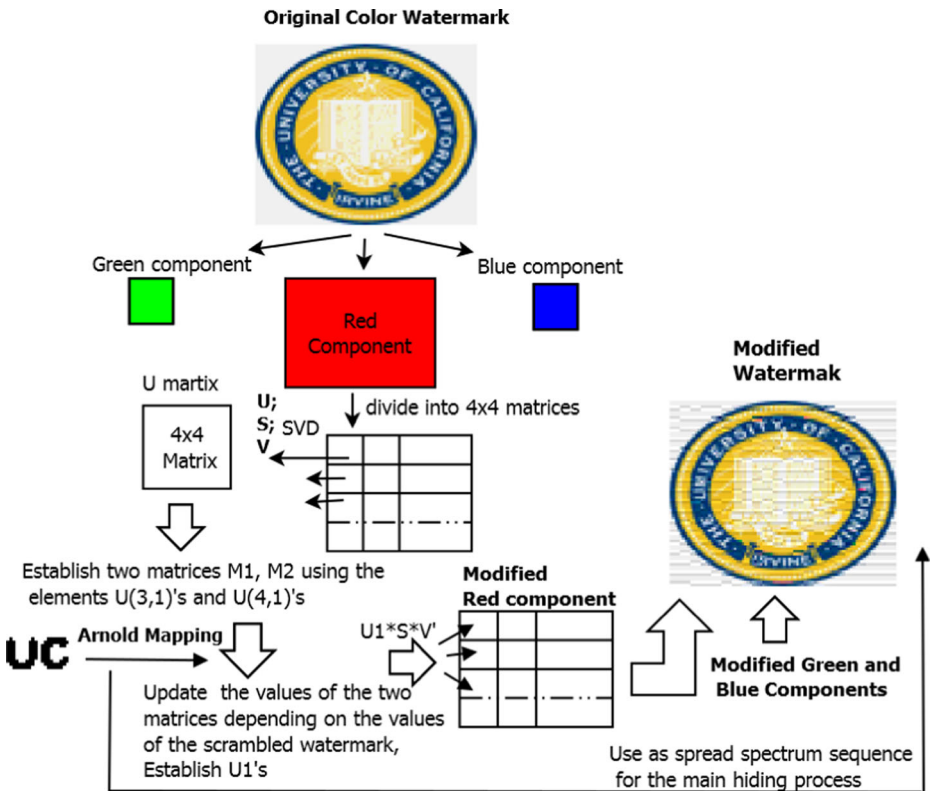


Fig. 9 Secondary hiding process

embedding process shown in Fig. 6. This requires of course a process of rearrangement and combining of the B&W watermarks and performing multiple Arnold Transforms, since the color watermark has larger size, and Arnold Transform is applied on square images only. This is shown in Fig. 10, where rearrangements and multiple Arnold Transforms are performed on our B&W watermark to get a bigger watermark that can be used for our main watermarking process.

The detection process for this secondary hiding process is the reverse of the hiding one. At the receive side and after extracting the bit-slices and the RGB components, SVD process is performed the same way shown in Fig. 9, then the 4x4 U matrices are established, and the elements  $u_{31}$  and  $u_{41}$  of each matrix are compared as shown in (8). Then reverse Arnold transform can be done to reconstruct the original watermark. This secondary hiding process has no effect on the quality of the original video since it modifies our color watermark only; so that it added to the security of the system without significant extra cost. Moreover, these watermarks have relatively small sizes, so any decompositions performed on them won't add too much to the computational complexity of the system.

$$W' = \begin{cases} 1, & \text{if } u_{31} > u_{41} \\ 0, & \text{if } u_{31} < u_{41} \end{cases} \tag{8}$$

## 4 Color watermark detection process

### 4.1 General extraction process

The extraction process is the reverse of the hiding process. The original video is not required, but still, the knowledge of the synthesis filter banks and the pseudorandom sequence is required. To extract the watermark, a prediction of the original values of the pixels is needed [21]. The watermarked image may be considered to be the original image that is disturbed by the pseudorandom noise. Due to the effect of the lossy compression, the additive noise and the numerous video processing operations that are being performed most of the time, the watermark detection process become a challenging one to overcome the false alarm detections on one hand, and the loss of hidden data on the other hand. Moreover, security arises as a critical issue that should be taken into consideration. The attacker would try to

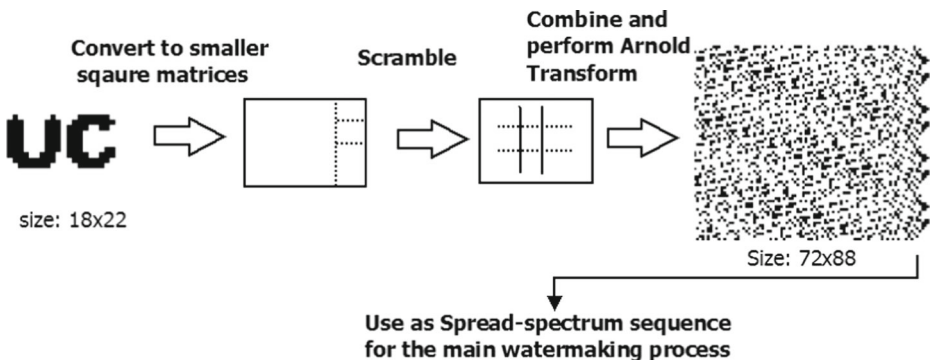


Fig. 10 Performing Arnold Transform on our B&W watermark

detect and extract the watermark or at least destroy it intentionally. This practice might result in the debilitation of the whole watermarking process and the loss of its effectiveness. To overcome these situations, an enhanced detection process is proposed. The detection process is aimed at extracting the binary bit-slices for further processing. The overall extracting process is shown in Fig. 11. The pseudo code for the extraction process is illustrated down.

**Reconstruction stage**

1. Read the incoming video stream  $F'_i: i=1,2,...n$
2. For  $i=1:n$  {
  - Get the YUV components  $F'_i \Rightarrow Y'_i, U'_i, V'_i$
  - Perform 1D Discrete Fourier Transform (1D-DFT) on  $Y'_i$  frames.
  - Perform Pyramid decomposition on the  $Y'_i$  frame
  - Perform one or two levels of DWT on the error image  $e_0$  based on the video resolution.
  - Choose the proper frame based on the (1D-DFT).
  - Choose the proper DWT band based on the Directive Contrast (DC).
  - Get the modified DWT coefficients, Denote them  $Q_j$  where  $j \in [1,12]$  based on the (1D-DFT).
  - Perform for each  $Q_j$  {
    - $Q'_j = \text{median filter}(Q_j)$
    - $L_j = Q'_j - Q_j;$
    - $LL_j = \text{sign}(L_j);$
    - $Q''_j = -1 * LL_j * P_j$

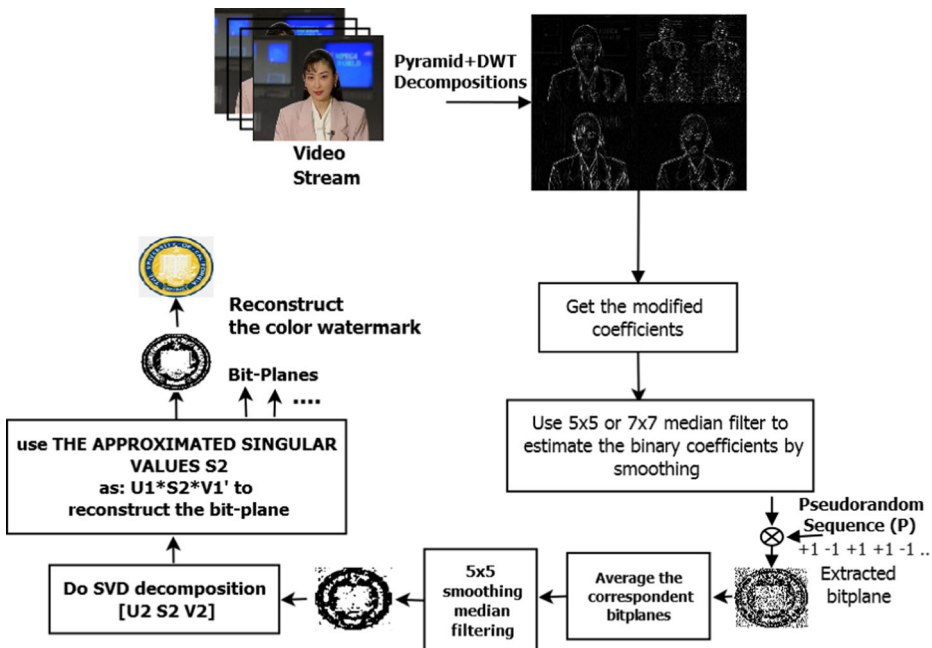


Fig. 11 The color watermark extraction process

```

Use 5X5 smoothing median filter (SMF):  $Q_j''' = SMF(Q_j'')$ 
Perform SVD process on  $Q_j'''$  to get the singular values:
 $[U_2j \ S2j \ V2j] = SVD(Q_j''')$ 
Use the approximated singular values  $S2j$  as:
 $U1j * S2j * V1j \Rightarrow$  reconstruct the bit-plane.
}end
    
```

3. Use the bit-planes to reconstruct the color watermark.

### 4.2 Enhanced detection process

A noise-elimination technique can be used to extract the hidden watermark pixels; to achieve that, a spatial convolution mask of size  $5 \times 5$  can be used to smoothen the extracted coefficients. Experimental results showed that the  $5 \times 5$  mask gave superior performance compared to the  $3 \times 3$  mask under different circumstances such as noise addition and compression processes. The enhanced detection process is then set to use multiple extracted watermarks, which were embedded in different video frames in the first place, for our final estimation process.

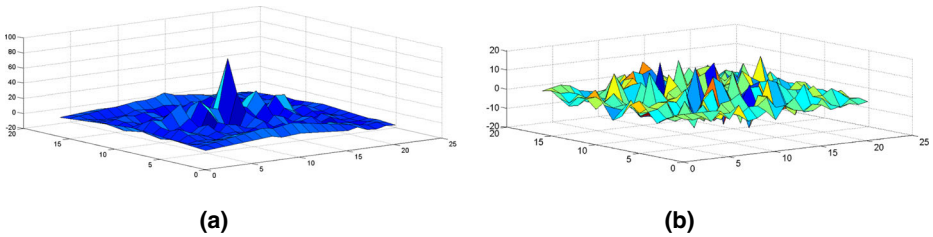
Let's assume that the extracted watermarks are grouped in a set  $W$  where  $W = w_1, w_2, \dots, w_n$ . To choose the set of watermarks that can be used in the final estimation process, cross-correlation test can be performed between every two extracted watermarks  $w_i$  and  $w_j$ . The cross-correlation between two matrices  $A$  and  $B$  is given according to (9):

$$R = \frac{\sum_m \sum_n (A_{mn} - \bar{A}) (B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2) (\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \tag{9}$$

where  $\bar{A}$  is the mean of  $A$  and  $\bar{B}$  is the mean of  $B$ . The cross correlation test is used primarily because of the fact that the extracted watermarks are the original watermarks that were embedded and thereafter corrupted due to the numerous video processing operations that were performed, and the intentional and unintentional attacks that the watermarked videos were subjected to. This can include geometric, statistical, and other types of attacks. Hence the extracted watermarks can be considered as noisy versions of the original ones, or in other words noisy signals. The cross correlation is a measure of similarity between two signals, and hence, the extracted watermarks would have a certain amount of similarity. Using this analogy, new set of extracted watermarks  $W_1$  can be established.

When the cross-correlation results in a significant peak at the center, this means that the two sets of extracted coefficients are useful and could be used for our final estimation process. Hence they could be included in the final watermarks set  $W_1$ . On the contrary, if the cross-correlation process did not result in significant peak at the center, this means that one or both of the sets of extracted coefficients are highly corrupted and hence one or both of them will be excluded from the final watermarks set. This can be shown in Fig. 12 where Fig. 12a shows a plot of the cross-correlation matrix between two extracted watermarks that are highly correlated and can be used in the final estimation process, while Fig. 12b shows the cross-correlation matrix between two extracted watermarks that one or the two of them are corrupted and therefore are discarded from the final estimation process. A threshold value for our estimation process is to be defined and set later depending on many factors such as the number of embedded watermarks and the intensity of the expected attacks.

The cross correlation between binary images as mentioned afore is a measure of similarity between them; this means that the flipping of any pixel would result in a reduction in the similarity parameter. If  $w_i \in W$ , then  $w_i$  is cross-correlated with all the other extracted



**Fig. 12** 3-D plots of the Cross-correlation matrices of two extracted watermarks

watermarks in the set, the average cross-correlation parameter is then computed. This process is repeated for the other watermarks in the set. A new set of average cross-correlation parameters is established that corresponds to each extracted watermark. By establishing a threshold  $h$  for the average cross-correlations, each extracted watermark that doesn't achieve the threshold test is excluded from the new set  $W_1$ . To get the final extracted watermark  $w_e$ , an averaging process is performed on the watermarks in the set  $W_1$ , where:

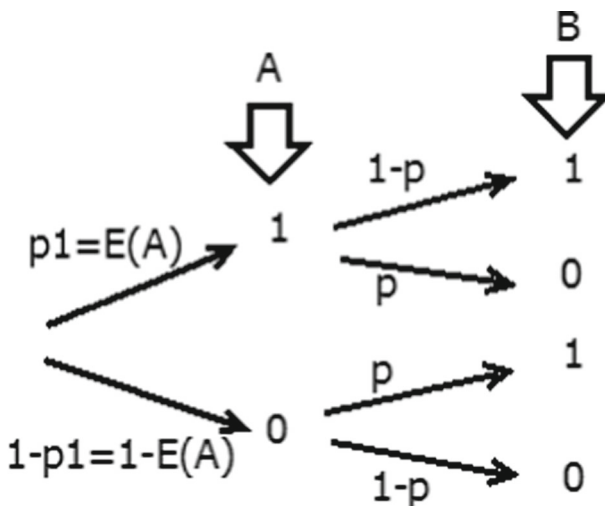
$$w_e = Ave\{W_1\} \tag{10}$$

Using the averaging process is attributed to statistical analysis. The correlation coefficient  $R$  between two matrices  $A$  and  $B$  which is used to measure the final performance is given in (9); the mean value of a binary image  $A$  is the expected value of  $A$  or  $E(A)$ . Assuming that the probability of 1's at the input is  $p_1$  and that the probability of flipping is  $p$  as shown in Fig. 13, and by taking into account that the average or mean value of a binary image is the expected value, then:

$$\bar{A} = E(A) = p_1 \tag{11}$$

Furthermore, the probability of getting 1 at the output  $\bar{B} = p_1 * (1 - p) + (1 - p_1) * p$ , and by taking (11) into account it can be written as:

$$\bar{B} = E(B) = E(A) + (1 - 2 * E(A)) * p \tag{12}$$



**Fig. 13** Expected values of the input and output binary images

Assuming that the input watermark  $A$  corresponds to a constant matrix during our watermarking process, the flipping probability of the pixels  $p$  is the only variable in the above equation. Furthermore, By comparing (9) and (12), it can be shown that the correlation between the two matrices  $A$  and  $B$  is a function of the flipping probability of the pixels, and hence by averaging the extracted watermarks, the flipping effect is eliminated to some extent and an enhanced version can be reconstructed as far as  $p$  is not equal to 0.5 which corresponds to an entropy value of one. To illustrate this analysis, Fig. 14 shows the relationships between these parameter when Gaussian noise with zero mean and different variances is applied to a random binary watermark; here the term density corresponds to the variance of noise for illustration purposes.

### 4.3 Estimating the singular values

The next step in our algorithm includes estimating the singular values that best approximate the original singular values for each slice. The singular values of the extracted watermark so far provided good approximation, but further processing is done to minimize the errors. The extracted bit-slices are smoothed by applying  $5 \times 5$  or  $7 \times 7$  median filtering on them, then SVD process is performed on each one. Figure 15 shows the approximated singular values for our  $72 \times 88$  watermark, the 8th slice of the red component. It can be seen that the error was minimized which helps in getting near perfect reconstruction.

Furthermore, Table 1 shows the Root-Mean-Square-Error (RMSE) values between the original singular values and the extracted and enhanced singular values that we've got using our approach. It can be shown that our approach enabled us to get singular values with RMSE's in the limits of 2.3. The enhancement after smoothing was at least 25%. It can be seen that the enhancement increases with the increase in the resolution of the videos; for HD (1920x1080) video, the enhancement reached 60%; this is expected since the larger resolution videos have more available spectral bands and hence more capacity of hiding in the DWT coefficients and this enabled us to have better establishment of the hidden watermark.

The proposed method can under no attacks circumstances to moderate attacks perform full reconstruction of the hidden data. The *stability of singular value theorem* indicates that, when there is a little disturbance with a matrix  $A$ , the variation of its singular value is not greater than 2-norm of disturbance matrix, where 2-norm is equal to the largest singular

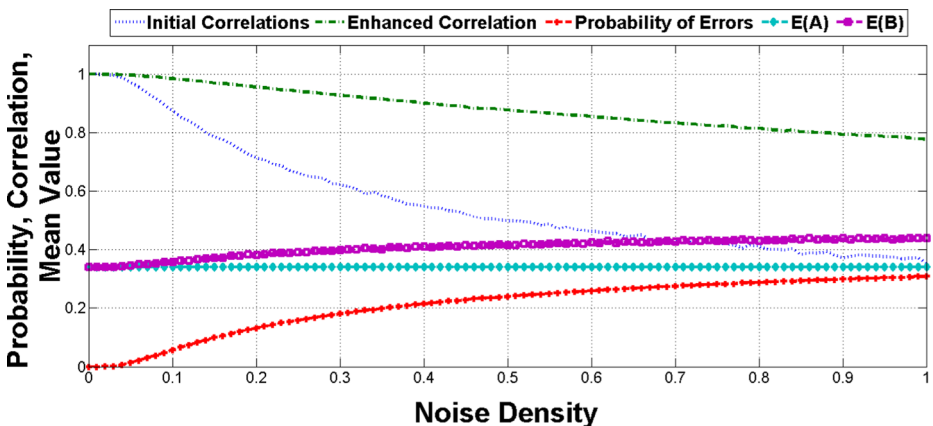


Fig. 14 The enhanced correlations vs. noise density



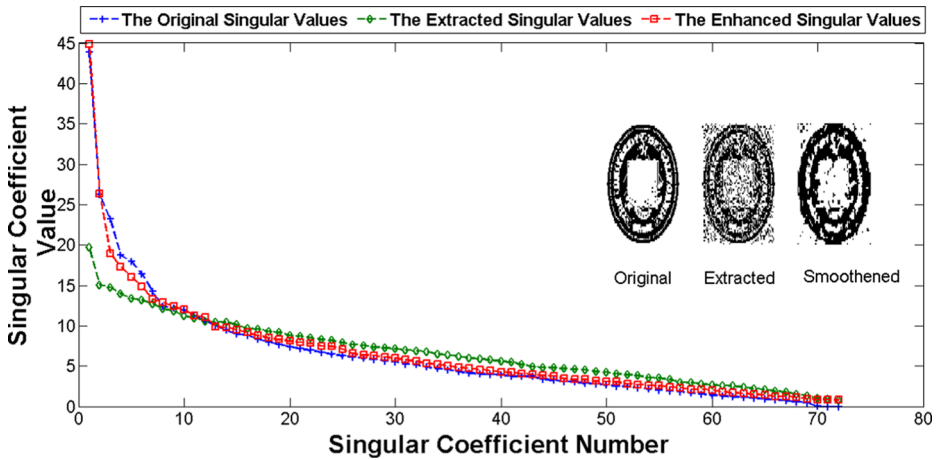


Fig. 15 Original and estimated Singular values of one Bit-Slice of the color watermark

value of the matrix [20]. From Fig. 15 and Table 1, it can be seen that by using our method, the variation in singular values were further minimized which would help in getting full reconstruction of the hidden watermark. The secondary B&W watermark can be extracted after getting the three RGB components as was shown in Section 3.

### 5 Experimental results

In this section we demonstrate the performance of our algorithm using our proposed method on different standard videos under different attacks. Furthermore, it will be compared with many proposed algorithms in the field. Watermarked and unwatermarked versions of a frame of *Mother-daughter* video ( $352 \times 288$  pixels) are shown in Fig. 16. The secondary B&W watermark of size  $18 \times 22$  and the original and modified color watermarks of size  $72 \times 88 \times 3$  are shown in Fig. 17. The secondary watermark and the extracted version are shown in Fig. 18. Furthermore, the modified hidden color watermark and the extracted one are shown in Fig.19.

#### 5.1 Visual quality and extraction process

Our algorithm performance will be evaluated in terms of PSNR between the original and the watermarked videos, and the Bit-Error-Rate (*BER*) between the original and the

Table 1 RMSE between original and extracted singular values

Video	RMSE (original and extracted)	RMSE (original and enhanced)	% Enhancement
Akiyo	3.16	2.35	25.6%
Foreman	3.37	2.44	27.6%
Mother-daughter	3.21	2.39	25.5%
BasketballDrill	4.03	2.52	37.4%
BasketballDrive	2.97	1.19	59.9%



Fig. 16 a Original mother-daughter frame, b watermarked frame

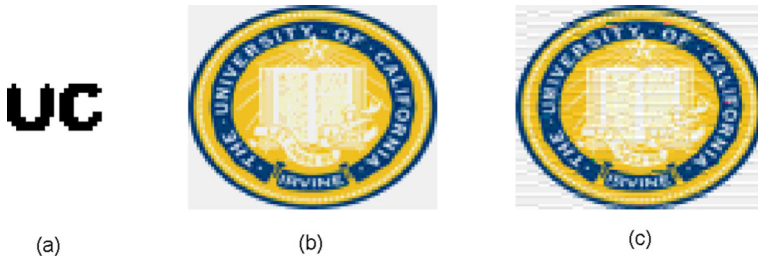


Fig. 17 a  $18 \times 22$  B&W original secondary watermark, b  $72 \times 88 \times 3$  Original color watermark c  $72 \times 88 \times 3$  Modified color watermark

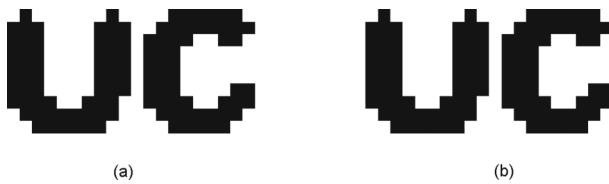


Fig. 18 a Original B&W secondary watermark, b The extracted watermark








Fig. 19 a Modified hidden color watermark b Extracted color watermark

extracted watermarks for the standard YUV videos: *Akiyo*, *Foreman*, *Mother-daughter*, *BasketballDrill* and *BasketballDrive*. The YUV space was selected for the raw videos because it is more efficient than the traditional RGB space and provides flexibility in choosing the resolutions required. For the CIF ( $352 \times 288$ ) videos, the watermark shown in Fig. 19 was used, while for the other two videos a color watermark of size  $60 \times 104 \times 3$  was used. The secondary watermarks are of sizes  $18 \times 22$  for the smaller CIF videos and  $15 \times 26$  for other videos. In these tests, multiple number of frames were watermarked. Moreover, since the metric measurements of the videos qualities are not always enough because the Human Visual System (HVS) perceives the quality in other ways, the Structural Similarity Index (SSIM) was evaluated for the watermarked frames of these videos. Table 2 shows these evaluated values for the videos. It can be seen that our proposed method achieved full reconstruction of the hidden color watermarks, moreover, the PSNR's were above 50 dB for all videos and sometimes over 60 dB which indicates good qualities. These good qualities were assured by the high values of the SSIM indices; all the SSIM indices were over 99% which indicates excellent visual perceptual quality and minimal distortions due to the hiding process.

Perceptual transparency test of the watermarked videos according to ITU-R Rec.500 and ITU-T Rec. P.800 and P.910 was performed using 25 people; this test is called the Mean Opinion Score (MOS), and it reflects how humans react to images and videos qualities, and how they evaluate them. The videos were shown to them without telling if the displayed video is watermarked or not. Table 3 shows the quality ratings of the videos; the table shows the number of individuals who gave their ratings for each video. It can be seen that most

**Table 2** The BER, PSNR and SSIM for the main watermarking process

Video stream	BER	PSNR	SSIM	Extracted watermark
Akiyo	0	54.4	0.9959	
Foreman	0	53.2	0.9967	
Mother-daughter	0	69.8	0.9985	
BasketballDrill	0	52.4	0.9994	
BasketballDrive	0	64.4	0.9997	

**Table 3** Mean Opinion Score (MOS) of the perceptual transparency of the watermarked videos (number of persons is 25)

Rating	5	4	3	2	1	Average
Original Akiyo	23	2	0	0	0	4.92
Watermarked Akiyo	22	3	0	0	0	4.88
Original Foreman	23	2	0	0	0	4.92
Watermarked Foreman	23	2	0	0	0	4.92
Original Mother-daughter	25	0	0	0	0	5
Watermarked Mother-daughter	24	1	0	0	0	4.96
Original BasketballDrill	23	2	0	0	0	4.92
Watermarked BasketballDrill	23	2	0	0	0	4.92
Original BasketballDrive	23	2	0	0	0	4.92
Watermarked BasketballDrive	22	3	0	0	0	4.88
Impairment	Impercep-tible	Perceptible, not annoying	Slightly annoying	Annoying	Very annoying	
Quality	Excellent	Good	Fair	Poor	Bad	

of the people did not notice any difference between the original and watermarked videos and they gave the same ratings for them. The evaluation process was almost unbiased, since the individual does not know if the shown video is watermarked or no. On the other hand, the human visual perceptual response to videos is a subjective issue; Table 4 shows the bias in this subjective test; it can be seen that the bias is very small in these kind of tests. Moreover, the table shows that the average perceptual MOS difference between the original and watermarked videos is 0.49%. Thus, the watermarking process is imperceptible; also, the qualities are mostly excellent.

Our watermarking process is a dual one, so the BER’s for the secondary watermarking process in the three components: the Red, the Green, and the Blue ones are shown in Table 5. It can be seen that the BER’s were zeros for the green and the blue components while they

**Table 4** The average MOS’s and the human visual bias for the standard test videos

Test videos	Optimal rating	Mean opinion score (MOS)	Percentage bias
Original videos	5	4.936	1.28%
Watermarked Videos	% change in MOS= $(4.936 - 4.9120)/4.936 = 0.49\%$		

**Table 5** The BER’s for the secondary hiding process in the three color components






Video stream	BER (red)	BER (green)	BER (blue)	Average %BER
Akiyo	0.0379	0	0	1.26
Foreman	0.0379	0	0	1.26
Mother-daughter	0.0379	0	0	1.26
BasketballDrill	0.0333	0	0	1.2
BasketballDrive	0.0333	0	0	1.2

were in the limits of 3.7% in the red component. The average values were in the limits of 1.2% which indicates high reliability of the system, especially that this is a secondary hiding process aimed at increasing the security of the overall system.

### 5.2 Robustness against attacks

To evaluate our proposed method, several attacks were performed to measure the degree of robustness of the system. The attacks that were used are: additive noise which include

**Table 6** Data-rates and the corresponding SSIM’s (the upper values) and PSNR’s (the lower values) of test videos after applying H.265 compression

Video stream	Bit-Rate: (kb/s)	SSIM PSNR (dB)	Bit-Rate: (kb/s)	SSIM PSNR (dB)	Bit-Rate: (kb/s)	SSIM PSNR (dB)	One frame with last Bit-Rate
Akiyo	1093	$\frac{0.9813}{49.4}$	75.5	$\frac{0.9479}{40}$	32.6	$\frac{0.7370}{30.7}$	
Foreman	2670	$\frac{0.9745}{45.7}$	162.0	$\frac{0.9068}{36.7}$	42.5	$\frac{0.7282}{28.3}$	
Mother-daughter	1426	$\frac{0.9873}{48.1}$	70.8	$\frac{0.9675}{40.9}$	29.4	$\frac{0.8124}{31.2}$	
Basketball-Drill	19444	$\frac{0.9532}{42.31}$	553.2	$\frac{0.8983}{36.0}$	85.5	$\frac{0.6847}{27.7}$	
Basketball-Drive	90963	$\frac{0.9679}{45.9}$	1389.2	$\frac{0.8970}{38.8}$	268.2	$\frac{0.7987}{31.6}$	

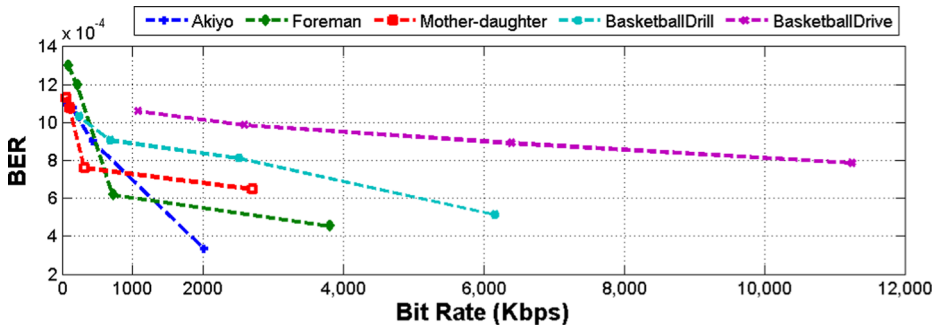


Fig. 20 BER of the hiding process when applying H.264 compression

(Gaussian, Poisson and salt-and-pepper noise), contrast adjustment, histogram equalization, median filter, rotation, frame dropping, frame averaging, frame swapping, transcoding, and H.264 and H.265 compressions. Moreover, the algorithm will be compared with other works in the field of video watermarking; it will be compared with the methods of [5, 17] and [30] under numerous attacks mentioned before.

Since the compression and transcoding attacks are among the most serious attacks that any video is subjected to, we will start our tests with them. The compressions being used are H.264 and H.265 under different data rates. High compression rates affect the perceptual quality of the videos; and since this work is concerned with protection of the ownership, so it would be useful to show the degradation in video quality with different compression rates before showing the algorithm performance. Two metrics will be used to show the quality; one is the Structural Similarity Index (SSIM), and the other is the PSNR while the compression being used is H.265. Table 6 shows the degradation in quality according to the data rates being achieved. One frame of each of these videos under the lowest data rate is shown to illustrate the downgraded quality. This table also shows that not always the objective tests are in line with the HVS perception of images or the subjective tests.

Figure 20 shows the performance of our system with the use of H.264, while Fig. 21 shows the performance of our system using the H.265. It can be shown from these two figures that our system is robust against compression process. Of course H.265 is more efficient than H.264, so that it can be seen that lower data rates in H.265 result almost in the same BER's as those of H.264.

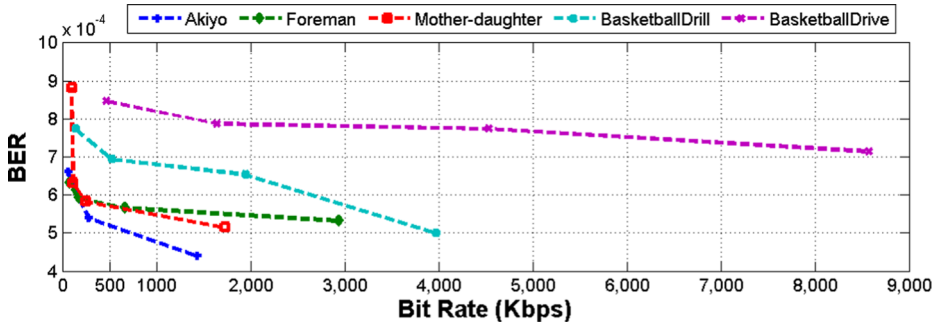


Fig. 21 BER of the hiding process when applying H.265 compression








**Fig. 22** The first frame of the watermarked mother-daughter video transcoded to: **a** AVI 832x480, **b** 3gp 480x320, **c** MP4 320x240

The transcoding process for videos is a common process in multimedia fields. Depending on the platform that the video is used in, numerous multi-media containers are used; some popular containers are:

- MP4: which is the standard audio and video container for the MPEG-4 multimedia portfolio.

**Table 7** The BER's and Data-rates of the color watermark extraction from different transcoded videos

Video stream	MP4, 320x240		AVI, 832x480		3gp, 480x320		Watermark
	BER	Bit-Rate: (kb/s)	BER	Bit-Rate: (kb/s)	BER	Bit-Rate: (kb/s)	
Akiyo	0.0299	201	0.0298	890	0.0362	349	
Foreman	0.0290	507	0.0285	2173	0.0359	895	
Mother-daughter	0.0290	247	0.0295	1199	0.0295	409	
Basketball-Drill	0.0179	548	0.0214	1920	0.0115	835	
Basketball-Drive	0.0103	501	0.0119	1914	0.0106	1171	

- 3GP: which is used by many mobile phones.
- AVI: which is the standard Microsoft Windows container.
















Of course this would result in many features such as variable frame rates, streaming capabilities and different data rates depending on the compression technique that is used and the format that is adopted. Moreover, Transcoding involves as well rescaling the videos to fit different display devices. Figure 22 shows the first frame of Mother-daughter video using the three formats: MP4, AVI and 3gp rescaled to different resolutions. The raw video is originally YUV 4:2:0 352x288. The watermarked videos were transcoded to these formats and rescaled, then the watermarks were extracted. Table 7 shows the BER's and the data rates for these videos. Also the extracted color watermarks are shown for the 3gp transcoding process. Table 8, however, shows the secondary hiding performance with the application of the transcoding processes. It can be seen that our dual hiding process is able to survive the transcoding attack with high reliability.

Our proposed method was tested under different other attacks and compared with methods [5, 17] and [30]; the attacks used were:

- Gaussian noise with variances 0.1 and 0.5.
- Poisson noise.
- Salt & pepper noise with 2% and 6% densities.
- Median filtering  $3 \times 3$  and  $5 \times 5$ .
- Contrast adjustment.
- Histogram equalization.










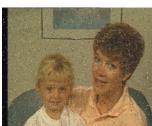








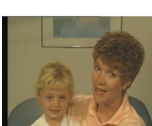





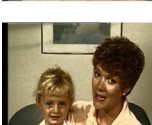


Table 9 shows our system performance using the *Mother-daughter* video for both the main and secondary watermarking processes when applying the aforementioned attacks, where BER1 and BER2 denote the bit error rates for the color and B&W watermarks respectively. Moreover, Fig. 23 shows the correlation values for the extraction process of the color

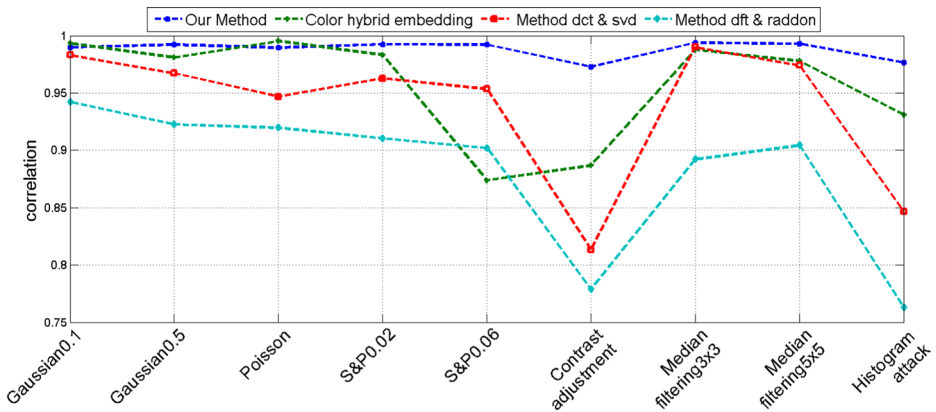
**Table 8** The BER's of the secondary watermark extraction from different transcoded videos

Video stream	BER: MP4, 320x240	Extracted watermark	BER: AVI, 832x480	Extracted watermark	BER: 3gp, 480x320	Extracted watermark
Akiyo	0.0589		0.0589		0.0623	
Foreman	0.0623		0.0598		0.0640	
Mother-daughter	0.0606		0.0614		0.0606	
BasketballDrill	0.0615		0.0615		0.0564	
BasketballDrive	0.0547		0.0573		0.0590	



**Table 9** BER's of our extraction method with several images processing attacks

Attack	Watermarked frame	BER1	Extracted watermark	BER2	Extracted watermark
Gaussian noise with variance 0.1		0.0114		0.0244	
Gaussian noise with variance 0.5		0.0115		0.0227	
Poisson noise		0.0112		0.0244	
Salt & pepper noise 2%		0.0096		0.0210	
Salt & pepper noise 6%		0.0114		0.0244	
Contrast adjustment		0.0051		0.0160	
Median filtering 3x3		0.0095		0.0202	
Median filtering 5x5		0.0111		0.0227	
Histogram equalization		0.0048		0.0160	



**Fig. 23** Correlations for the proposed extraction process and methods ‘Color hybrid embedding [5]’, ‘dct&svd [30]’, ‘dft&raddon [17]’ for the color watermarks under some common attacks

watermarks under those common attacks using our method and other methods. It can be seen that our method provides almost the best correlation values of all the other methods. Furthermore, our method was able to provide high correlation values; in fact more than 97% for all the attacks.

Geometrical attacks are ones of the most aggressive and serious attacks in multimedia watermarking fields. One of them which is the scaling attack was addressed in the videos transcoding beforehand; the other geometrical attack is the rotation attack; it involves rotating all or some of the video frames around the center point for specific degrees. Most traditional transform watermarking methods fail to survive this attack unless some sort of readjustment and realignment were applied before the extraction process. Because of the *rotation invariant property* of the singular values [20], it is possible to survive this attack with good robustness. Our watermarked videos were subjected to several rotations (1, 2, 5, 10 and 180 degrees), and then our extraction process was applied as mentioned beforehand. Figure 24 shows the BER’s at these rotations for both our method and method of [5] which has the best BER’s of all other methods. It can be seen that our method outperformed the method in [5] for all the assumed angles of rotations.

The other attacks that are common in video watermarking are the temporal attacks. From their name, it’s clear that temporal attacks are more related to multiple frames rather than one frame which in turn can be called space attacks, and which we’ve addressed so far. Common temporal attacks are: *Frame Dropping*, *Frame Swapping*, and *Frame Inserting*. Since the swapping attack involves inserting process, so we will test our method against frame dropping and frame swapping attacks only. Different numbers of frames were dropped from the standard videos that were used, and afterwards, the average BER’s of our extraction process were evaluated. Our results were compared with the method in [5] which outperformed other methods; here, the average BER’s that were achieved in the method of [5] were used; these results are shown in Fig. 25. Likewise, different numbers of frames were swapped over the course of running our watermarked videos. In fact, we expect that this attack is less aggressive than the dropping attack, were whole frames are lost and hence important hidden data are lost as a result. Here in frame swapping attacks, the successive video frames especially before scene changing are highly correlated and as a consequence, in general, the hidden data are lost partially only. Figure 26 shows the performance of our method as

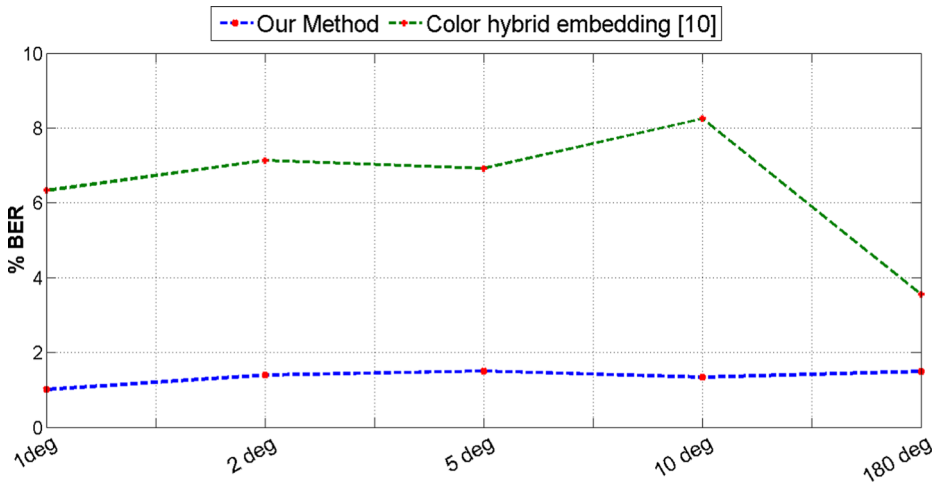


Fig. 24 BER’s of the extraction process under several rotation attacks

well as the method in [5] in terms of BER’s. For both attacks, our method outperformed the method in [5].

To further demonstrate our system performance, Table 10 shows the performance of our system for both the color watermark and the secondary watermark under specific geometric and temporal attacks for the video of *mother-daughter*. Here the rotation was 10 degrees counter-clockwise, 20% of the frames were dropped randomly for the dropping attack, and 33% of the frames were swapped randomly for the swapping attack. Moreover, cropping attacks were performed for randomly chosen frames, where 25% of the frame was cropped from either the left-upper or the right-lower sides. It can be seen that our system is able to survive these attacks with high reliability in both the main and the secondary watermarking processes. Here for the frame dropping attack, the BER’s depend mainly on the length of the video and hence the total number of frames that are watermarked; this means that the performance can be enhanced dramatically with practical real life videos rather than the test videos which tend to be short.

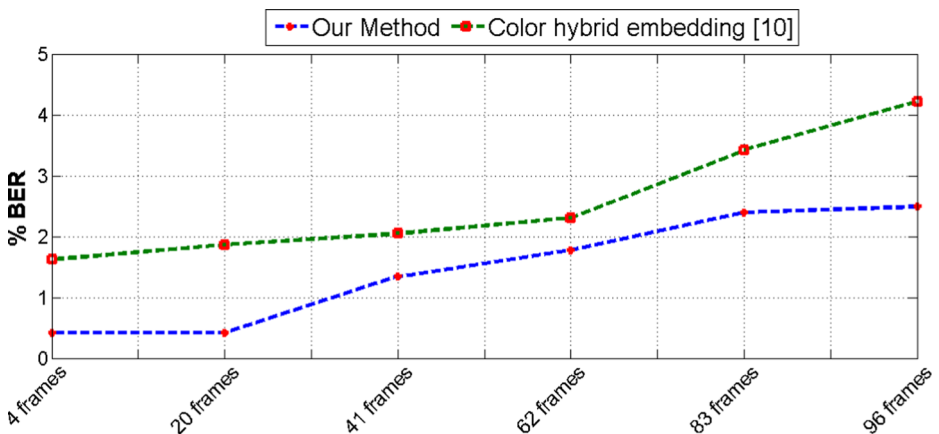


Fig. 25 BER’s of the extraction processes under different frame dropping attacks

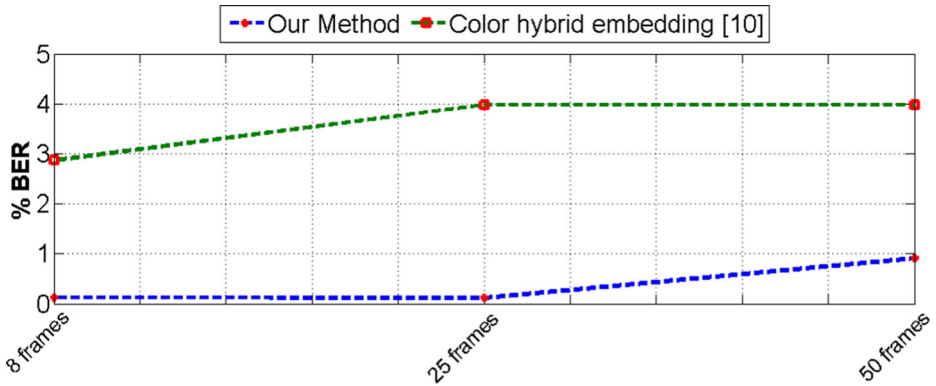


Fig. 26 BER’s of the extraction processes under different frame swapping attacks

Our proposed technique, moreover, was tested against averaging attack. We discussed the collusion attacks and the ways they could be implemented in Section 2; furthermore, we introduced our method of using the 1D DFT to find the static and dynamic frames. Our

Table 10 BER’s of our method with specific geometric and temporal attacks
















Attack	Watermarked frame	BER1	Extracted watermark	BER2	Extracted watermark
Rotation (10 deg)		0.0111		0.0253	
Frame dropping (20%)		0.0218		0.0463	
Frame swapping (33%)		0.0078		0.0177	
Frame Cropping (25%)		0.0168		0.0438	
Frame cropping (25%)		0.0146		0.0320	

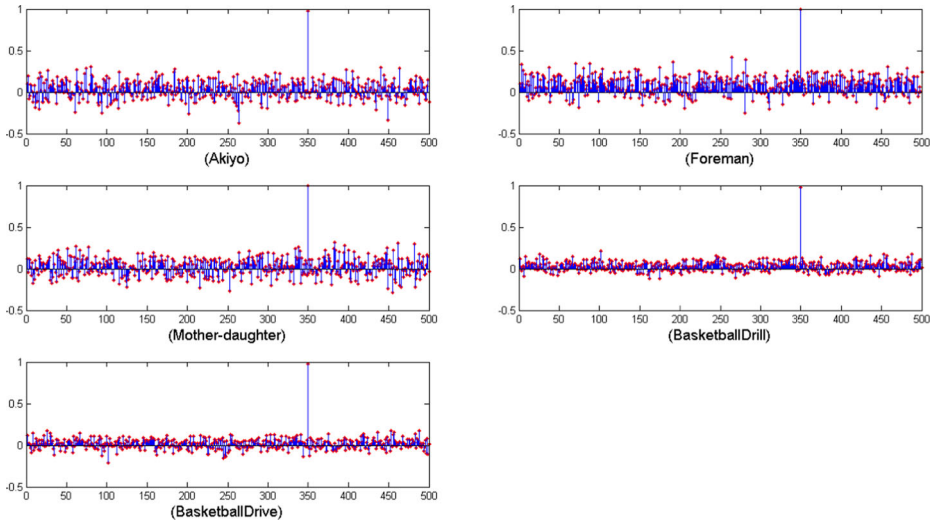
data hiding scheme was performed using these frames accordingly. In fact, frame averaging is considered a collusion attack aimed at removing the hidden watermark. We performed frames averaging in the same scenes of the videos to see the results; Table 11 shows our results for the five standard videos, and the BER's for both the main watermarking process (BER1) and for the secondary watermarking process (BER2), and the extracted watermarks. It can be seen that our dual watermarking process was able to survive the averaging attack with high reliability.

To ensure the security of the scheme, the watermarking process was tested for false alarms attacks. That's when the system indicates the existence of the inserted watermark while, in fact, no watermark was embedded or another watermark was the one that was hidden actually. For the test videos, 500 different generated random B&W watermarks were embedded and the right watermark was set to be the 200th one. The results are shown in Fig. 27. It can be shown that our system responded with low correlations to all the embedded watermarks except the 200th one, which indicates a high reliability from this aspect.

Furthermore, it's useful to check the detection performance of the system. The detection process is more concerned with true or false detections of the presence of a watermark in an object, rather than extracting the whole watermark. The watermark embedding process is repeated for at least 50 times for each test video, by using different watermark sequences

**Table 11** BER's of our method with frames averaging attacks

Attack	Four averaged frames	BER1	Extracted watermark	BER2	Extracted watermark
Akiyo		0.0054		0.0185	
Foreman		0.0068		0.0202	
Mother-daughter		0.0069		0.0185	
Basketball-Drill		0.0042		0.0214	
Basketball-Drive		0.0039		0.0197	



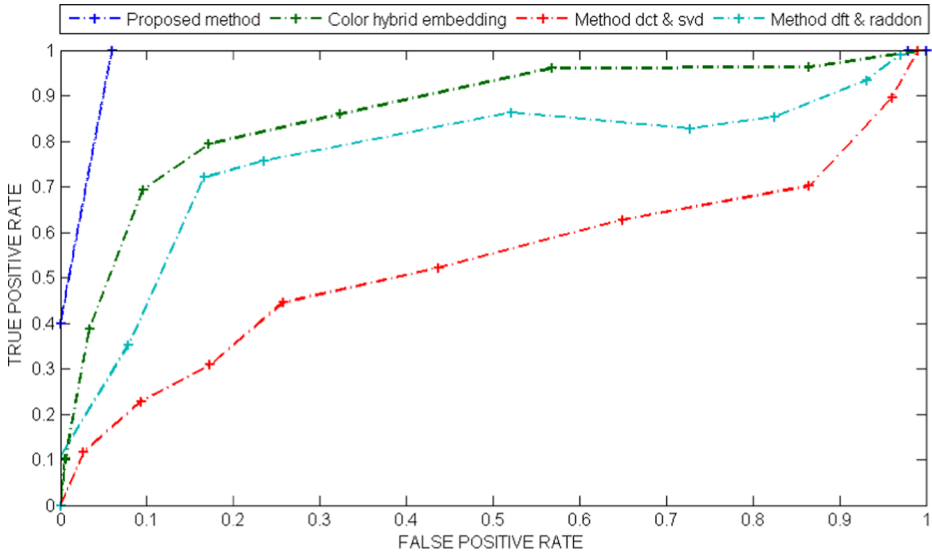
**Fig. 27** The watermarking process response to false alarm test, the right watermark is the 200th

and keys, and under aggressive attacks; then the Receiver Operating Characteristics (ROC) curve is plotted. This curve is a plot of the probability of true positive detection versus the probability of false positive detection [12]. The false positive detection occurs when the detector detects a watermark in an unwatermarked object, which is in this case the video sequence; while the true positive detection happens when the detector detects the presence of a watermark in a watermarked object. Figure 28 shows the detection performance of our method under aggressive attacks (Gaussian noise with variance 0.1, H.265 compression (QP=25), Median Filtering 3x3); moreover, the ROC curves of other methods of video watermarking are shown as well. Here the ROC curves are built on BER's values rather than correlations, and the average BER's for the test videos is used. It can be seen that our method is superior to other methods in the detection performance.

### 5.3 Computational complexity of the dual hiding process

To perform the watermarking in the pyramid and the wavelet transforms, two requirements should be met. First, the filter banks should be generated randomly; this means that the decomposition structure and the bands being used for watermarking must be determined by the owner in the hiding stage. The second requirement for practical watermarking system is to perform the hiding and the extracting processes in minimum time. Storage requirements are small; the filters can be generated by changing the coefficients only. The running time is related directly to the computational complexity of the Pyramid, DWT and SVD processes.

Computational complexity for the Pyramid and DWT processes depends on the number of operations (here multiplications) required to transform an image for a number of levels  $N$ . A mathematical derivation of this complexity for pyramid transform is introduced in [2]. The derivation assumes that circular convolution based on Fast Fourier Transform (FFT) and Inverse Fast Fourier Transform (IFFT) is used for transforming an image pyramidally. It is well known that the overall complexity of conducting convolution for  $B$  points via the FFT is:  $O(B \log_2 B)$  which is lower than  $O(B^2)$  of the computation of  $B$ -point Discrete



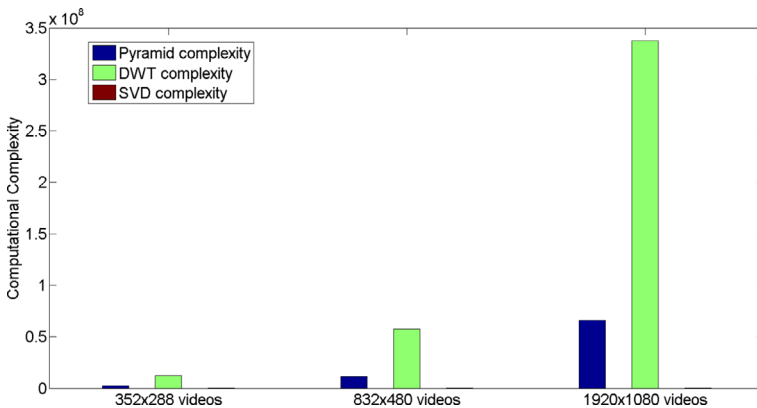
**Fig. 28** ROC curves for our method and other methods 'Color hybrid embedding [5]', 'dct&svd [30]', 'dft&raddon [17]'

Fourier Transform (DFT). The derivation is summarized below for an input image  $x_0$ ; the pyramid decomposition for this image was shown in Fig. 2.

For an image  $x_0(n_1, n_2)$  of size  $L_1 * L_2$ , the number of multiplications needed for the first level  $x_1(n_1, n_2)$  with decimation factor  $M$  will be:

$$(L_1 L_2) * \log_2(L_1) + \frac{L_2 L_1}{M} * \log_2(L_2) \tag{13}$$

The first part of (13) results from horizontal filtering and the second part is the number of multiplications needed for vertical filtering after a decimation by  $M$ . Let  $K_1 = L_1 \log_2 L_1$ , and  $K_2 = L_2 \log_2 L_2$ , then (13) can be applied for higher levels. In general, the total number of multiplications needed to get the decimated images  $x_1(n_1, n_2), x_2(n_1, n_2)$ ,



**Fig. 29** The computational complexities for the three decompositions being used, the pyramid, the DWT, and the SVD for one video frame

...,  $x_{N-1}(n_1, n_2)$  and the difference images  $e_0(n_1, n_2), e_1(n_1, n_2), \dots, e_{N-2}(n_1, n_2)$  can be written as follows [9]:

$$2 \left[ L_2 K_1 + \frac{L_1}{M} K_2 \right] \quad \text{for } N = 1 \tag{14}$$

$$2 \left[ \sum_{i=0}^{N-1} \left[ \frac{L_2 K_1}{M^{2i}} + \frac{L_1 K_2}{M^{2i+1}} \right] - L_1 L_2 \sum_{i=0}^{N-2} \left[ (i+1) \frac{M+1}{M^{2i+3}} \right] \right] \quad \text{for } N \geq 2 \tag{15}$$

where  $N$  is number of decomposition levels,  $M$  is the decimation factor which is generally 2; moreover, the hiding and extracting stages are taken into account. The above analysis can be extended to the wavelet transform taking into account that there are four filters for each stage of decompositions and four filters for each stage of reconstructions, and the decimation factor is  $M=2$ . Number of multiplications in the wavelet transform is shown in (16) and (17).

$$8 \left[ L_2 K_1 + \frac{L_1}{M} K_2 \right] \quad \text{for } N = 1 \tag{16}$$

$$8 \left[ \sum_{i=0}^{N-1} \left[ \frac{L_2 K_1}{M^{2i}} + \frac{L_1 K_2}{M^{2i+1}} \right] - L_1 L_2 \sum_{i=0}^{N-2} \left[ (i+1) \frac{M+1}{M^{2i+3}} \right] \right] \quad \text{for } N \geq 2 \tag{17}$$

The singular value decomposition, on the other hand, has a computational complexity of  $O(\min\{mn^2, nm^2\})$  [9], where  $n$  and  $m$  are the dimensions of the matrix that is being decomposed. As was shown in Section 3, the RGB color watermark was converted to its three components, the red, the green and the blue; then each component was divided to  $4 \times 4$  matrices, where the SVD process is performed. According to the complexity limit of the SVD process, the computational complexity is  $4 \times 4^2 = 64$  for each matrix. If the size of the watermark is  $72 \times 88$ , then the total complexity in the hiding process is  $(72 \times 88/16) \times 3 \times 64 = 76032$  and the same for the extraction process. The computational complexities for the three decompositions being used, the pyramid, the DWT, and the SVD for one video frame are shown in Fig. 29; moreover, the computational complexities in Log scale are shown in Fig. 30. Here the pyramid and DWT processes correspond to the main hiding process, while the SVD process corresponds to the secondary hiding process. The increase in the computational complexity due to the use of the secondary hiding process is 0.5%, 0.1% and 0.02% for the videos from lower to higher resolutions respectively. This increase is

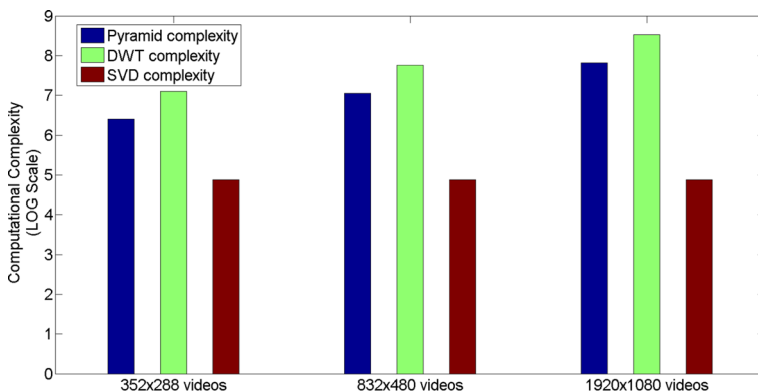


Fig. 30 The computational complexities in Log scale for the three decompositions being used, the pyramid, the DWT, and the SVD for one video frame



marginal given the increase in the security that was achieved due to our dual hiding process. Moreover, the computational complexities evaluated here did not include every aspect of the operations that were performed during our hiding process, which mostly took place in the main hiding process rather than the secondary one; we are more concerned with the limiting behavior of the operations  $O(\text{operations})$ , and the increase due to the duality in our hiding process.

## 6 Conclusions and future work

This paper proposes a dual hybrid Pyramid-Wavelet-SVD watermarking process using Gaussian filters and randomly generated orthonormal filter banks. The hiding process is a dual one that uses two different hiding processes. The purpose of this duality is to increase the security of the system without significant cost in the computational complexity on one hand or the visual quality of the original videos on the other hand. Our main color watermark method does not require the original videos for the extraction process like most of the SVD-based watermarking techniques. The singular values were estimated and approximated using the enhanced detection process. The errors in the singular values were reduced to the point that perfect reconstructions of our watermarks were achieved. Furthermore, the RGB color watermark was split into Bit-Plane slices of binary images for the hiding process; the binary spaces are more efficient in hiding when it comes to security, estimation and extraction; this would enable us to use other means of security such as spread spectrum sequences. Other color spaces for the watermark such as YUV could've been used in the same manner. The B&W watermark matrix partition SVD-based hiding was proposed in a previous work, but we used it as a secondary hiding in the color watermark. This didn't affect the visual quality of the original frames, and at the same time, the Arnold transformed image was used as a random sequence for our main color watermarking process. Our method proved to be robust against the H.264 and H.265 compression attacks and the Transcoding attack. Moreover, it proved to be robust and superior to previous work under the traditional image processing attacks, geometrical attacks and temporal attacks. The visual quality which is achieved is excellent since it was possible to control the weighting factor of the hiding process. Throughout the course of the hiding, extracting and estimating processes, it was possible to reduce the weighting factor to the point that the visual qualities were maintained at more than 99% for the SSIM parameter, and the PSNR values above 55 dB on average. Different other helpful techniques were used to increase the robustness of the system such as the *directive contrast*, *1D-DFT*, adaptive hiding, and smoothening processes. The computational complexity analysis of the dual hiding process proved that the increase in complexity was marginal due to this duality. Furthermore, the detection process is excellent and superior to other methods in the field. Future work will investigate the extension of this work to the 3D videos.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## References

1. Abdallah EE, Hamza AB, Bhattacharya P (2010) Video watermarking using wavelet transform and tensor algebra. *Signal Image Video Process* 4(2):233–245

2. Al-Asmari AK (1995) Optimum bit rate pyramid coding with low computational and memory requirements. *IEEE Trans Circuits Syst Video Technol* 5(3):182–192
3. Al-Asmari AK, Al-Enizi FA (2009) A pyramid-based watermarking technique for digital color images copyright protection. In: International conference on computing, engineering and information ICC'09. IEEE, pp 44–47
4. Agarwal P, Kumar A, Choudhary A (2015) A secure and reliable video watermarking technique. In: 2015 International Conference on computer and computational sciences (ICCCS). IEEE, pp 151–156
5. Agilandeewari L, Ganesan K (2016) A robust color video watermarking scheme based on hybrid embedding techniques. *Multimed Tools Appl* 75(14):8745–8780
6. Bhattacharya S, Chattopadhyay T, Pal A (2006) A survey on different video watermarking techniques and comparative analysis with reference to h. 264/avc. In: ISCE'06 IEEE Tenth international symposium on consumer electronics, 2006. IEEE, pp 1–6
7. Bhatnagar G, Raman B (2009) A new robust reference watermarking scheme based on DWT-SVD. *Comput Standards Interfaces* 31(5):1002–1013
8. Burt P, Adelson E (1983) The Laplacian pyramid as a compact image code. *IEEE Trans Commun* 31(4):532–540
9. Frieze A, Kannan R, Vempala S (2004) Fast Monte-Carlo algorithms for finding low-rank approximations. *J ACM (JACM)* 51(6):1025–1041
10. Guzmán VVH, Miyatake MN, Meana HMP (2004) Analysis of a wavelet-based watermarking algorithm. In: 14th International conference on electronics, communications and computers CONIELECOMP 2004. IEEE, pp 283–287
11. Kundur D, Hatzinakos D (1998) Digital watermarking using multiresolution wavelet decomposition. In: Proceedings of the IEEE international conference on acoustics, speech and signal processing, 1998, vol 5. IEEE, pp 2969–2972
12. Kutter M, Petitcolas FA (1999) Fair benchmark for image watermarking systems. In: Electronic imaging'99. International Society for Optics and Photonics, pp 226–239
13. Lee M-S (2003) Image compression and watermarking by wavelet localization. *Int J Comput Math* 80(4):401–412
14. Li X, Niu J, Khan MK, Liao J (2013) An enhanced smart card based remote user password authentication scheme. *J Netw Comput Appl* 36(5):1365–1371
15. Liao X, Qin Z, Ding L (2017) Data embedding in digital images using critical functions. *Signal Process Image Commun* 58:146–156
16. Liao X, Yin J, Guo S, Li X, Sangaiiah AK (2018) Medical jpeg image steganography based on preserving inter-block dependencies. *Comput Electrical Eng* 67:320–329
17. Liu Y, Zhao J (2010) A new video watermarking algorithm based on 1d dft and radon transform. *Signal Process* 90(2):626–639
18. Martinez R, Reyes R, Cruz C, Nakano M, Perez H (2008) A dwt-based video watermarking scheme resilient to mpeg-2 compression and collusion attacks. In: ISITA International symposium on information theory and its applications, 2008. IEEE, pp 1–5
19. Meerwald P, Uhl A (2001) Survey of wavelet-domain watermarking algorithms. In: Photonics West 2001-electronic imaging. International Society for Optics and Photonics, pp 505–516
20. Mohan BC, Kumar SS (2008) A robust image watermarking scheme using singular value decomposition. *J Multimed* 3(1):7–15
21. Panyavaraporn J (2011) Wavelet based video watermarking scheme for h. 264/avc. In: 2011 International symposium on intelligent signal processing and communications systems (ISPACS). IEEE, pp 1–5
22. Panyavaraporn J (2013) Multiple video watermarking algorithm based on wavelet transform. In: 2013 13th International Symposium on communications and information technologies (ISCIT). IEEE, pp 397–401
23. Potdar VM, Han S, Chang E (2005) A survey of digital image watermarking techniques. In: 2005 3rd IEEE International conference on industrial informatics, 2005 INDIN'05. IEEE, pp 709–716
24. Su Q, Niu Y, Zou H, Liu X (2013) A blind dual color images watermarking based on singular value decomposition. *Appl Math Comput* 219(16):8455–8466
25. Sullivan GJ, Ohm J-R, Han W-J, Wiegand T (2012) Overview of the high efficiency video coding (hevc) standard. *IEEE Trans CircuitsSyst Vid Technol* 22(12):1649–1668
26. Sullivan GJ, Boyce JM, Chen Y, Ohm J-R, Segall CA, Vetro A (2013) Standardized extensions of high efficiency video coding (hevc). *IEEE J Selected Topics Signal Process* 7(6):1001–1016
27. Vetterli M (1995) J. kova cevi c wavelets and subband coding
28. Vinod P, Bora P (2005) A new inter-frame collusion attack and a countermeasure. In: International workshop on digital watermarking. Springer, pp 147–157

29. Wang S-H, Lin Y-P (2004) Wavelet tree quantization for copyright protection watermarking. *IEEE Trans Image Process* 13(2):154–165
30. Wu X, Sun W (2013) Robust copyright protection scheme for digital images using overlapping dct and svd. *Appl Soft Comput* 13(2):1170–1182
31. Xu D, Wang R, Wang J (2008) Video watermarking based on spatio-temporal jnd profile. In: *International Workshop on digital watermarking*. Springer, pp 327–341



**Farhan Alenizi** received his PhD in Electrical Engineering and Computer Science from the University of California, Irvine in 2017. He received his BSc and MSc degrees in Electrical Engineering from King Saud University in 1999 and 2006 respectively. His areas of interest are images and video processing, signal processing, Discrete Signal Processing (DSP), images and video watermarking, 3d mesh objects watermarking, secure multimedia exchanges. He has been a faculty member at the department of Electrical Engineering at PSAU University in Saudi Arabia since 2010. He is now an assistant professor at this university. Besides his academic experience, he worked for 10 years at the Saudi Telecom Company (STC), a leading telecommunication company in the Middle East as a designer and consultant in the Satellite and optical fiber transmission networks.



**Fadi Kurdahi** received his PhD from the University of Southern California in 1987. Since then, he has been a faculty at the Department of Electrical & Computer Engineering at UCI, where he conducts research in the areas of Computer Aided Design and design methodology of large scale systems. He serves as the Associate Dean for Graduate and Professional Studies of the Henry Samueli School of Engineering, and the Director of the Center for Embedded & Cyberphysical Systems (CECS), comprised of world-class researchers in the general area of Embedded and Cyber-physical Systems. He served on numerous editorial boards, and was program chair or general chair on program committees of several workshops, symposia and conferences in the area of CAD, VLSI, and system design. He received the best paper awards for the *IEEE Transactions on VLSI* in 2002, *ISQED* in 2006 and *ASP-DAC* in 2016, and other distinguished paper awards at *DAC*, *EuroDAC*, *ASP-DAC* and *ISQED*. He also received the Distinguished Alumnus award from his *Alma Mater*, the American University of Beirut in 2008. He is a Fellow of the *IEEE* and the *AAAS*.



**Ahmed M. Eltawil** has been a faculty member with the Department of Electrical Engineering and Computer Science, at the University of California, Irvine since 2005, where he is currently Professor and Director of the Wireless Systems and Circuits Laboratory. His research interests are in the general area of low power digital circuit and signal processing architectures with an emphasis on mobile systems. In addition to his department affiliation, he is also affiliated to a number of research centers across the University of California, Irvine. He received the Doctorate degree from the University of California, Los Angeles, in 2003 and the M.Sc. and B.Sc. degrees (with honors) from Cairo University, Giza, Egypt, in 1999 and 1997, respectively. Dr. Eltawil has been on the technical program committees and steering committees for symposia, and conferences in the areas of low power computing and wireless communication system design. He received several awards, as well as distinguished grants, including the NSF CAREER grant supporting his research in low power systems.



**Awad Kh. Al-Asmari** was born in Balasmer, Saudi Arabia (SA). He received his B.Sc from King Saud University, Riyadh, SA, and his MSc from Ohio State University, Columbus, U.S.A., and his PhD from the University of Toledo, Ohio, U.S.A. All in the area of communication and signal processing. Since 1991 he has been a faculty member at the College of Engineering, King Saud University, SA, where he rose through the ranks to become a full professor in 2003. In addition to this, he was appointed Vice Rector for Graduate Studies and Research at Prince Sattam Bin Abdulaziz University in Al-Kharj, SA. He then promoted to be parliament member for 4 years. Now, he is the president of Shaqra University, SA. Prof. Awad Alasmari is deeply involved in numerous services to his immediate constituencies, both geographical and professional. He has been a senior member of the IEEE since 1996 and has co-authored numerous publications in the areas of image processing, information security, telecommunications, nanotechnology and quantum computing. He can be reached at [alasmari@su.edu.sa](mailto:alasmari@su.edu.sa).

## Affiliations

Farhan Alenizi<sup>1</sup>  · Fadi Kurdahi<sup>2</sup> · Ahmed M. Eltawil<sup>2</sup> · Awad Kh. Al-Asmari<sup>3</sup>

Fadi Kurdahi  
kurdahi@uci.edu

Ahmed M. Eltawil  
aeltawil@uci.edu

Awad Kh. Al-Asmari  
alsruoor@gmail.com

<sup>1</sup> Prince Sattam bin Abdulaziz University, Al-Kharj, Saudi Arabia

<sup>2</sup> Center for Embedded and Cyber-Physical Systems, University of California, Irvine (UCI), Irvine, CA, USA

<sup>3</sup> Shaqra University, Shaqra, Saudi Arabia